PHP в облаках





Обо мне

- Больше 10 лет в разработке
- Руководил разработкой и поддержкой крупного СМИ про киберспорт
- Сейчас в Авито, пишу бэкенд



О чём поговорим

- Зачем в облако?



О чём поговорим

- Зачем в облако?
- Как изменится приложение



О чём поговорим

- Зачем в облако?
- Как изменится приложение
- Разные окружения



О чём поговорим

- Зачем в облако?
- Как изменится приложение
- Разные окружения
- Состав пода



О чём поговорим

- Зачем в облако?
- Как изменится приложение
- Разные окружения
- Состав пода
- Метрики/логи



О чём поговорим

- Зачем в облако?
- Как изменится приложение
- Разные окружения
- Состав пода
- Метрики/логи
- CI/CD





One time I tried to explain Kubernetes to someone.

Then we both didn't understand it.

16:40 · 06/08/2019 · Twitter for iPhone



А зачем в облако? ⛅



Зачем?

- Утилизация серверов



Зачем?

- Утилизация серверов
- Rolling update и откат



Зачем?

- Утилизация серверов
- Rolling update и откат
- Масштабирование



Зачем?

- Утилизация серверов
- Rolling update и откат
- Масштабирование
- Отказоустойчивость



Зачем?

- Утилизация серверов
- Rolling update и откат
- Масштабирование
- Отказоустойчивость
- Разные стеки



Допилим приложение 



Как изменится приложение

- Статические файлы
- Внешние ресурсы
- Сессии
- Консольные команды
- Миграции
- Конфигурация



12 factor app

- Зависимости
- Конфигурация
- Сторонние службы
- Сборка, релиз, выполнение
- Утилизируемость
- Паритет окружений разработки и прода



Статические файлы

- Ассеты
- Загруженные файлы
- Изображения



Ассеты

- В образ
- CDN после сборки



Uploads

- Не хранить в папке
- Хранилище отдельно от приложения
- Внешнее (s3)
- Внутреннее (ceph или иное)



Изображения

- Похоже на файлы
- Прокси для ресайза, вотермарок
- Imgproxy, thumbor
- Удобно!



Внешние ресурсы

- База данных
- Кэш
- Сессии



База данных

- Прод — лучше отдельно
- Тестовые — лучше в кубах



Кэш

- In-memory
- В каждом инстансе
- Общий для всех инстансов



Сессии

- По умолчанию лежат в файлах, это не подходит
- Выносим сессии в redis/memcache/БД



Крон и прочие консольные команды

- Нужен мьютекс
- Есть в symfony и laravel
- Job и CronJob в k8s
- Или отдельным контейнером, если хочется, например, laravel scheduler



Миграции

- Та же проблема с одновременным запуском
- Можно в CI
- Можно как Job и InitContainer
- Database migration with k8s Job and InitContainers



Конфигурация

- Env!
- При запуске, а не сборке
- Особенности фреймворков и их DI



Секреты

- K8s secrets
- Vault
- Github/gitlab variables
- Можно при старте контейнера складывать в env



Логи

- Не пишем в файл
- Отправляем в stdout/stderr
- Можно JSON
- log_errors_max_len в php.ini по умолчанию 1024 байт



Приложение готово

- Статика вынесена отдельно
- БД и кэш — отдельно
- Сессии не в файлах
- Консольные команды и крон в отдельном контейнере
- Миграции в InitContainer
- Конфигурация в env



Среды 🐸



Окружения

- Production/staging/local
- Разные внешние ресурсы
- FPM/CLI
- Внутренние зависимости отличаются (phpunit, xdebug)



composer.lock

- В репозиторий и в образ
- Осознанно меняем версии библиотек



Xdebug

- Нужен локально, а на проде? Стейдже?
- Но есть оверхед…или нет?
- Тред с обсуждением



Xdebug



Derick Rethans
@derickr

В ответ @dr4goonis @sam_dark и @asgrim

There is also no runtime overhead if you set xdebug.mode=off when using Xdebug 3



Dockerfile.basic

- Его будем катить на прод
- Общие настройки (timezone, модули PHP)
- COPY –from=composer:latest /usr/bin/composer /usr/local/bin/composer
- composer install —no-dev
- Уменьшаем размер, значит alpine и multistage build



Dockerfile.stage

- composer install —dev
- PHPUnit
- Xdebug



Dockerfile.dev

- Xdebug (если не сделали раньше)
- xhprof
- Composer
- Можно запускать в Minikube или docker-compose
- Монтируем локальную папку с исходниками



Образы — итог

- Один образ, скорее всего, не получится, но это ок
- Basic (prod)
- PHP-FPM/CLI
- Stage/dev
- Можно объединять некоторые образы



Состав пода 📦



Pod

- Pod — набор запускаемых в k8s контейнеров
- Один экземпляр приложения
- Горизонтальное масштабирование
- Containers & InitContainers
- Контейнеров может быть один или несколько



Состав типичного пода

- Приложение (PHP-FPM)



Настройка PHP-FPM

- listen = 127.0.0.1:9000
- pm=static
- pm.max_children=10
- pm.max_requests=1000
- pm.status_path=/_info
- ping.path=/healthz



Состав типичного пода

- Приложение (PHP-FPM) слушает порт
- Веб-сервер (nginx)
- Pgbouncer
- Twemproxy/haproxy
- Сбор метрик/логов (php-fpm_exporter)



Как работает? 👀



Наблюдаем за работой

- Liveness probe — проверяем, что доступен /healthz



Наблюдаем за работой

- Liveness probe — проверяем, что доступен /healthz
- Readiness probe — можно проверить, что есть БД и кэш, можно прогреть кэши



Наблюдаем за работой

- Liveness probe — проверяем, что доступен /healthz
- Readiness probe — можно проверить, что есть БД и кэш, можно прогреть кэши
- Логи собираем в elasticsearch/graylog



Наблюдаем за работой

- Liveness probe — проверяем, что доступен /healthz
- Readiness probe — можно проверить, что есть БД и кэш, можно прогреть кэши
- Логи собираем в elasticsearch/graylog
- Метрики push — можно в readiness probe
- Метрики pull — prometheus



CI/CD 🛠️



CI/CD — как было

- git pull
- composer install
- migrate up
- Меняем symlink



CI/CD — как теперь

- Нужен container registry
- Build
- Статический анализ и линтеры
- Тесты
- Деплой/откат по кнопке



CI/CD — деплой

- Готовый образ
- Миграции
- Прогрев кэша (например, readiness пробой)
- Rolling update



Итог 🤘



Итог

- Код соответствует 12 факторам
- Окружения продуманы, образы описаны
- Под содержит всё нужное
- Собираем логи и метрики
- Деплоим и откатываем по нажатию на кнопку
- Масштабируем, спокойно обновляем сервера



Что дальше?

- Для теста можно направить часть трафика в кластер
- Следим за метриками, тюним FPM и лимиты
- Horizontal Pod Autoscaling (HPA)
- Можно сделать тестовые среды для фичей



Чеклист переезда в облако

https://cutt.ly/phprussia-cloud





Полезные материалы

- https://12factor.net/ru/
- Статья про заталкивание монолита Авито в k8s
- Запуск одного экземпляра консольного приложения в Laravel
- Статья и доклад про сетевой обмен в микросервисах на PHP
- Тред с обсуждением оверхеда от xdebug



Обратная связь и комментарии по докладу по ссылке





@Paaashka



